UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/029,497	12/21/2001	Sivaram Krishnan		4638

7590 05/04/2005

David T. Cunningham, Esq.
Hitachi America, Ltd.
50 Prospect Avenue
Tarrytown, NY 10591

EXAMINER	
GUILL, RUSSELL L	

ART UNIT	PAPER NUMBER
2123	

DATE MAILED: 05/04/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/029,497	KRISHNAN, SIVARAM	
	Examiner	Art Unit	
	Russell L. Guill	2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 December 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) _____ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1 - 31* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *21 December 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date [illegible] 1 page
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1 – 31 have been examined. Claims 1 – 31 have been rejected.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2.1. Claim 22 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The claim recites, "said emulator". The phrase appears to have insufficient antecedent basis. For the purpose of claim examination, the phrase, "said emulator", is interpreted as "said emulation sequence generator". Correction or amendment is required.

2.2. Claim 24 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The claim recites, "said media". The phrase appears to have insufficient antecedent basis. For the purpose of claim examination, the phrase, "said media", is interpreted as "said physical implementation". Correction or amendment is required.

2.3. Claims 30 - 31 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Both claims recite, "an emulation code generator generating the emulated sequence of instructions executable a first

instruction set from the original sequence of instructions executable with a different second instruction set". The examiner does not understand the phrase. For the purpose of claim examination, the phrase is interpreted as "an emulation code generator generating the emulated sequence of instructions, replacing a first instruction set from the original sequence of instructions with a different second instruction set". Correction or amendment is required.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 1 - 10 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The claims appear to be entirely algorithmic, and though executed by a computer, produce no tangible result.

5. Claims 11 - 15 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The claims appear to not necessarily include hardware, and are therefore intangible. For the purpose of claim examination, the phrase "a computer readable storage media having computer readable code implementing", is interpreted as "a computer readable storage media on which are recorded instructions that cause a processor to execute".

6. Claims 17 - 31 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The claims appear to be entirely algorithmic, and though executed by a computer, produce no tangible result.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

8. Claims 1 – 3, 6, 10 – 13, 17 – 18, 22 are rejected under 35 U.S.C. 102(b) as being anticipated by Altman (Altman, Erik R.; Kaeli, David; Sheffer, Yaron; "Welcome to the opportunities of Binary Translation", March 2000, IEEE Computer).

8.1. Regarding claims 1 and 11, Altman appears to teach a method to be performed by a computer system having at least one computer, storage media for physical implementation of an emulated sequence of instructions of the computer and translated from an original sequence of instructions ***(page 40 – 41, section labeled "Three Types of Translation"; and page 44, left-side column, paragraph that starts with "Translated applications . . .")***, and at least one component that produces dynamic execution information ***(page 41, section labeled "Profiling")***, said

method improving performance of the emulated sequence of instructions ***(page 41, section labeled "Dynamic Optimization")*** and comprising the steps of:

8.1.1. Producing first dynamic execution information in response to executing the emulated sequence of instructions ***(page 41, section labeled "Profiling", first paragraph)***; and

8.1.2. Changing the computer system for producing different dynamic execution information in response to said first dynamic execution information ***(page 41, section labeled "Profiling", second paragraph, second sentence)***.

8.2. Regarding claims 2 and 12, Altman appears to teach modifying at least parameters of instructions of the emulated sequence of instructions ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***.

8.2.1. Regarding ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***, the registers are parameters of instructions as defined in the specification on page 9, lines 5 – 15.

8.3. Regarding claims 3 and 13, Altman appears to teach modifying at least register fields of instructions of the emulated sequence of instructions ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***.

8.4. Regarding claim 6, Altman appears to teach providing the original sequence of instructions and generating the emulated sequence of instructions from the original

sequence of instructions ***(page 40, title; and abstract directly beneath the title; and left-side column, paragraph 3, definition of "binary translation")***.

8.5. Regarding claim 10, Altman appears to teach that changing generates a modified emulated sequence of instructions by modifying at least some instructions of the emulated sequence of instructions in response to at least some of the dynamic execution information ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item and second bullet item)***.

8.5.1. Regarding ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item and second bullet item)***; both the "ISA remapping" and "basic block reordering", modify instructions.

8.6. Regarding claims 11, 12, and 13, Altman appears to teach a computer readable storage media having computer readable code implementing a method for improving performance of an emulated sequence of instructions ***(pages 40 – 41)***.

8.6.1. Regarding ***(pages 40 – 41)***; dynamic binary translation is performed using a computer, and a computer inherently has computer readable storage media having computer readable code.

8.7. Regarding claims 17 and 18, Altman appears to teach:

8.7.1. A node to provide an original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation" – please note that it was inherent that a dynamic translator has a node to provide an original sequence of instructions)***

8.7.2. Means for generating the emulated sequence of instructions from the original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

8.7.3. An input to receive the execution information produced from execution of the emulated sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

8.7.3.1. Regarding ***(pages 40 – 41, section labeled "Three types of Translation")***; it would have been inherent that a memory or CPU registers receive the execution information produced from execution of the emulated sequence of instructions.

8.8. Regarding claim 22, Altman appears to teach:

8.8.1. A computer coupled to said emulator ***(pages 40 – 41, section labeled "Three Types of Translation")***.

8.8.1.1. Regarding ***(pages 40 – 41, section labeled "Three Types of Translation")***; it would have been inherent that the emulator and dynamic translator are coupled to a computer.

8.8.2. Storage media coupled to said computer ***(pages 40 – 41, section labeled "Three Types of Translation")***;

8.8.2.1. Regarding ***(pages 40 – 41, section labeled "Three Types of Translation")***; it would have been inherent that the storage media are coupled to the computer (e.g. RAM).

8.8.3. An emulator generating an emulated sequence of instructions from the original sequence of instructions ***(pages 40 – 41, section labeled "Three Types of Translation"; and page 40, left-side column, third paragraph)***;

8.8.4. Means for producing the dynamic execution information from execution of the emulated sequence of instructions ***(page 42, section labeled "Profiling"; and page 42, section labeled "Collaborative Profiling")***.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman (Altman, Erik R.; Kaeli, David; Sheffer, Yaron; "Welcome to the opportunities of Binary Translation", March 2000, IEEE Computer) in view of Kelly (U.S. Patent 5,832,205).

10.1. Claim 4 is a dependent claim of claim 1, and thereby inherits all of the rejected limitations of claim 1.

10.2. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

10.3. The art of Kelly is directed to translating machine instructions for one computer into machine instructions to run on a second computer ***(column 11, lines 54 – 67)***.

10.4. Altman does not specifically teach software producing multiple conditions codes that replace a single condition code of the first dynamic execution information.

10.5. Kelly appears to teach software ***(column 14, lines 50 – 60)*** producing multiple conditions codes that replace a single condition code of the first dynamic execution information ***(figures 6(a) – (c); and column 14, lines 3 – 60)***.

10.5.1. Regarding ***(figures 6(a) – (c); and column 14, lines 3 – 60)***; it would have been obvious that in the VLIW processor of figure 6(c), that a single condition code of figure 6(b) is replaced by multiple condition codes, since the VLIW processor contains multiple copies of the functional hardware units such as an INT ALU, allowing parallel execution of machine instructions which would have been executed serially on the RISC processor using a single functional hardware unit.

10.6. The art of Kelly and the art of Altman are analogous art because they are both directed to the problem of translating machine instructions for one computer into machine instructions to run on a second computer.

10.7. The motivation to use the art of Kelly with the art of Altman would have been obvious given the benefit recited in Kelly of providing a host processor with apparatus for enhancing the operation of a microprocessor which is less expensive than conventional state of the art microprocessors yet is compatible with and capable of running application programs and operating systems designed for other microprocessors at a faster rate than those other microprocessors ***(column 9, lines 20 – 29)***.

10.8. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Kelly with the art of Altman to produce the claimed invention.

11. Claims 5, 14, 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Altman as applied to claim 1 above, in view of Lethin (Patent Application Publication Number US 2002/0147969).

11.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

11.2. The art of Lethin is directed to translating machine instructions for one computer into machine instructions to run on a second computer ***(paragraph [0001])***.

11.3. Regarding claim 14, Altman appears to teach a computer readable storage media having computer readable code implementing a method for improving performance of an emulated sequence of instructions ***(pages 40 – 41)***.

11.3.1. Regarding ***(pages 40 – 41)***; it would have been obvious that dynamic binary translation is performed using a computer, and it would have been obvious that a computer has computer readable storage media having computer readable code.

11.4. Regarding claim 19, Altman appears to teach:

11.4.1. A node to provide an original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation" – please note that it would have been obvious that a dynamic translator has a node to provide an original sequence of instructions)***

11.4.2. Means for generating the emulated sequence of instructions from the original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

11.4.3. An input to receive the execution information produced from execution of the emulated sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

11.4.3.1. Regarding ***(pages 40 – 41, section labeled "Three types of Translation")***; it would have been obvious that a memory or CPU registers receive the execution information produced from execution of the emulated sequence of instructions.

11.5. Regarding claim 5, Altman does not specifically teach that the steps of executing, producing, and changing are conducted recursively on at least some of successive segments of the emulated sequence of instructions.

11.6. Regarding claim 5, Lethin appears to teach that the steps of executing, producing, and changing are conducted recursively on at least some of successive segments of the emulated sequence of instructions ***(paragraphs [0381] and [0384])***.

11.7. The art of Lethin and the art of Altman are analogous art because they are both directed toward translating machine instructions for one computer into machine instructions to run on a second computer.

11.8. The motivation to use the art of Lethin with the art of Altman would have been obvious because of the benefit recited in Lethin that the invention improves performance of the emulation relative to template-based translations and template-based interpretations ***(paragraph [0054])***.

11.9. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Lethin with the art of Altman to produce the claimed invention.

12. Claims 7, 15, 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Altman as applied to claim 1 above, in view of Conte (Conte, Thomas M.; Patel, Burzin A.; Cox, J. Stan; "Using Branch Handling Hardware to Support Profile-Driven Optimization", 1994, Proceedings of the 1994 27th annual international symposium on microarchitecture).

12.1. Claim 7 is a dependent claim of claim 1, and thereby inherits all of the rejected limitations of claim 1.

12.2. Claim 15 is a dependent claim of claim 7, and thereby inherits all of the rejected limitations of claim 7.

12.3. Claim 20 is a dependent claim of claim 15, and thereby inherits all of the rejected limitations of claim 15.

12.4. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

12.5. The art of Conte is directed to using branch handling hardware to support profile-driven optimization ***(page 1, Title)***.

12.6. Regarding claim 15, Altman appears to teach a computer readable storage media having computer readable code implementing a method for improving performance of an emulated sequence of instructions ***(pages 40 – 41)***.

12.6.1. Regarding ***(pages 40 – 41)***; it would have been obvious that dynamic binary translation is performed using a computer, and it would have been obvious that a computer has computer readable storage media having computer readable code.

12.7. Regarding claim 20, Altman appears to teach:

12.7.1. A node to provide an original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation" – please note that it was inherent that a dynamic translator has a node to provide an original sequence of instructions)***

12.7.2. Means for generating the emulated sequence of instructions from the original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

12.7.3. An input to receive the execution information produced from execution of the emulated sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

12.7.3.1. Regarding ***(pages 40 – 41, section labeled "Three types of Translation")***; it would have been obvious that a memory or CPU registers receive the execution information produced from execution of the emulated sequence of instructions.

12.8. Regarding claim 7, Altman does not specifically teach that the step of producing produces branch prediction information.

12.9. Regarding claim 7, Altman does not specifically teach that the step of changing changes condition codes of the branch prediction information.

12.10. Regarding claim 7, Conte appears to teach that the step of producing produces branch prediction information ***(pages 3 – 4, section 3 Using Branch Prediction Hardware for Profiling; and page 2, section 2 Branch Prediction and Profiling)***.

12.11. Regarding claim 7, Conte appears to teach that the step of changing changes condition codes of the branch prediction information ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***.

12.11.1. Regarding ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***; the state is updated which changes the state condition code.

12.12. The art of Conte and the art of Altman are analogous art because they both contain the problem of profiling and optimization.

12.13. The motivation to use the art of Conte with the art of Altman would have been obvious given the benefit recited in Conte that using the specified branch prediction method produces a dramatic effect in achieving 96% accuracy in branch prediction ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph)***;

12.14. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Conte with the art of Altman to produce the claimed invention.

13. Claims 8, 16, 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Altman (Altman, Erik R.; Kaeli, David; Sheffer, Yaron; "Welcome to the opportunities of Binary Translation", March 2000, IEEE Computer) in view of Wall (Wall, David W.; "Global Register

Allocation at Link Time", October 28, 1986, www.hpl.hp.com/techreports/Compaq-DEEC/WRL-86-3.html).

13.1. Claim 8 is a dependent claim of claim 1, and thereby inherits all of the rejected limitations of claim 1.

13.2. Claim 16 is a dependent claim of claim 8, and thereby inherits all of the rejected limitations of claim 8.

13.3. Claim 21 is a dependent claim of claim 16, and thereby inherits all of the rejected limitations of claim 16.

13.4. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

13.5. The art of Wall is directed to optimizing register allocation, including using profile information to perform optimization ***(page 1, Abstract)***.

13.6. Regarding claim 16, Altman appears to teach a computer readable storage media having computer readable code implementing a method for improving performance of an emulated sequence of instructions ***(pages 40 – 41)***.

13.6.1. Regarding ***(pages 40 – 41)***; it would have been obvious that dynamic binary translation is performed using a computer, and it would have been obvious that a computer has computer readable storage media having computer readable code.

13.7. Regarding claim 21, Altman appears to teach:

13.7.1. A node to provide an original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation" – please note that it was inherent that a dynamic translator has a node to provide an original sequence of instructions)***

13.7.2. Means for generating the emulated sequence of instructions from the original sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

13.7.3. An input to receive the execution information produced from execution of the emulated sequence of instructions ***(pages 40 – 41, section labeled "Three types of Translation")***

13.7.3.1. Regarding ***(pages 40 – 41, section labeled "Three types of Translation")***; it would have been obvious that a memory or CPU registers receive the execution information produced from execution of the emulated sequence of instructions.

13.8. Regarding claim 8, Altman does not specifically teach that the step of producing produces a history of register allocation information.

13.9. Regarding claim 8, Altman does not specifically teach that the step of changing changes register allocation.

13.10. Regarding claim 8, Wall appears to teach producing a history of register allocation information ***(page 1, Abstract, paragraph 3; and page 12, section 2.4.4. Profiling)***.

13.11. Regarding claim 8, Wall appears to teach changing register allocation ***(page 1, Abstract, paragraph 3; and page 12, section 2.4.4. Profiling)***.

13.12. The art of Wall and the art of Altman are analogous art because they both include the problem of profiling a program execution and using the profile information to optimize the program.

13.13. The motivation to use the art of Wall with the art of Altman would have been obvious because of the benefit recited in Wall that benchmark program speedup was 10 to 25 percent ***(page 1, Abstract, paragraph 3)***.

13.14. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Wall with the art of Altman to produce the claimed invention.

14. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman (Altman, Erik R.; Kaeli, David; Sheffer, Yaron; "Welcome to the opportunities of Binary Translation", March 2000, IEEE Computer) in view of Conte (Conte, Thomas M.; Patel, Burzin A.; Cox, J. Stan; "Using Branch Handling Hardware to Support Profile-Driven Optimization", 1994, Proceedings of the 1994 27th annual international symposium on microarchitecture).

14.1. Claim 9 is a dependent claim of claim 1, and thereby inherits all of the rejected limitations of claim 1.

14.2. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

14.3. The art of Conte is directed to using branch handling hardware to support profile-driven optimization ***(page 1, Title)***.

14.4. Altman does not specifically teach that the step of producing produces a history of branch dynamic execution information.

14.5. Altman does not specifically teach that the step of changing generates a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group.

14.6. Conte appears to teach that the step of producing produces a history of branch dynamic execution information ***(pages 3 – 4, section 3 Using Branch Prediction Hardware for Profiling; and page 2, section 2 Branch Prediction and Profiling)***.

14.7. Conte appears to teach that the step of changing generates a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***.

14.7.1. Regarding ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***; it would have been obvious that the step of changing generates a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group.

14.8. The art of Conte and the art of Altman are analogous art because they both contain the problem of profiling and optimization.

14.9. The motivation to use the art of Conte with the art of Altman would have been obvious given the benefit recited in Conte that using the specified branch prediction method produces a dramatic effect in achieving 96% accuracy in branch prediction ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph)***;

14.10. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Conte with the art of Altman to produce the claimed invention.

15. Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman as applied to claim 22 above, in view of common knowledge in the art.

15.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

15.2. Altman does not specifically teach a computer system according to claim 22 further comprising:

15.2.1. An additional computer; and

15.2.2. A network interconnecting said computer, said additional computer, said storage media and said means for producing.

15.3. At the time of invention, it was old and extremely well known in the art to interconnect multiple computer systems by way of a network in order to improve resource sharing. It would have been obvious to the ordinary artisan at the time of invention to use the art of Altman in combination with his own knowledge of the art to arrive at the claimed invention.

16. Claim 24 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman (Altman, Erik R.; Kaeli, David; Sheffer, Yaron; "Welcome to the opportunities of Binary Translation", March 2000, IEEE Computer) in view of common knowledge in the art.

16.1. Altman appears to teach a computer ***(pages 40 – 41, section labeled "Three Types of Translation")***.

16.1.1. Regarding ***(pages 40 – 41, section labeled "Three Types of Translation")***; it would have been obvious that a dynamic binary translator is executed on a computer.

16.2. Altman appears to teach a means for providing a physical implementation of an emulated sequence of instructions of said computer and produced from an original sequence of instructions ***(pages 40 – 41, section labeled "Three Types of Translation")***.

16.2.1. Regarding ***(pages 40 – 41, section labeled "Three Types of Translation")***; it would have been obvious that a physical implementation of the emulated sequence of instructions is produced (e.g. RAM).

16.3. Altman appears to teach a means for producing dynamic execution information in response to execution of an emulated sequence of instructions ***(page 41, section labeled "Profiling")***.

16.4. Altman appears to teach a means for responding to the dynamic execution information and for changing the computer system so that at least some dynamic execution information obtained on subsequent execution of the emulated sequence of instructions would be changed ***(page 41, section labeled "Dynamic optimization")***.

16.5. Altman does not specifically teach a coupling between a computer and media suitable to provide for execution of the emulated sequence of instructions on at least the computer.

16.6. At the time of invention, it was old and extremely well known in the art to have a coupling between a computer and media to provide execution of instructions (e.g. a CPU and RAM), and indeed, such coupling is built into a computer. It would have been obvious to the ordinary artisan at the time of invention to use the art of Altman in combination with his own knowledge of the art to arrive at the claimed invention.

17. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman and common knowledge in the art, as applied to claim 24 above, in view of Conte (Conte, Thomas M.; Patel, Burzin A.; Cox, J. Stan; "Using Branch Handling Hardware to Support Profile-Driven Optimization", 1994, Proceedings of the 1994 27th annual international symposium on microarchitecture).

17.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

17.2. The art of Conte is directed to using branch handling hardware to support profile-driven optimization ***(page 1, Title)***.

17.3. Altman does not specifically teach maintaining a record of branch addresses in the emulated sequence of instructions historically correlated to whether branches were taken during execution of the emulated sequence of instructions.

17.4. Altman does not specifically teach means for changing a likelihood condition code of the branch prediction information for at least one of the branches.

17.5. Conte appears to teach means for maintaining a record of branch addresses in the emulated sequence of instructions historically correlated to whether branches were

taken during execution of the emulated sequence of instructions ***(page 2, section 2 "Branch Prediction and Profiling"; and figure 1; and figure 2)***.

17.6. Conte appears to teach means for changing a likelihood condition code of the branch prediction information for at least one of the branches ***(page 2, section 2 "Branch Prediction and Profiling"; and figure 1; and figure 2)***.

17.7. The art of Conte and the art of Altman are analogous art because they both contain the problem of profiling and optimization.

17.8. The motivation to use the art of Conte with the art of Altman would have been obvious given the benefit recited in Conte that using the specified branch prediction method produces a dramatic effect in achieving 96% accuracy in branch prediction ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph)***;

17.9. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Conte with the art of Altman to produce the claimed invention.

18. Claims 26 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Altman and common knowledge in the art, as applied to claim 24 above.

18.1. Regarding claim 26, Altman appears to teach modifying at least parameters of instructions of the emulated sequence of instructions ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***.

18.1.1. Regarding ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***, the registers

are parameters of instructions as defined in the specification on page 9, lines 5 – 15.

18.2. Regarding claim 27, Altman appears to teach modifying at least register fields of instructions of the emulated sequence of instructions ***(page 41, section labeled "Dynamic Optimization", second paragraph, first bullet item that starts with "ISA remapping")***.

19. Claims 28 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Altman and common knowledge in the art, as applied to claim 24 above, in view of Wall (Wall, David W.; "Global Register Allocation at Link Time", October 28, 1986, www.hpl.hp.com/techreports/Compaq-DEEC/WRL-86-3.html).

19.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

19.2. The art of Wall is directed to optimizing register allocation, including using profile information to perform optimization ***(page 1, Abstract)***.

19.3. Regarding claim 28, Altman does not specifically teach cycling allocation of registers in a pool of registers as some of successively identified registers in the emulated sequence of instructions.

19.4. Regarding claim 29, Altman does not specifically teach producing a history of temporary register allocation information.

19.5. Regarding claim 29, Altman does not specifically teach changing register parameters of the emulated sequence of instructions.

19.6. Regarding claim 28, Wall appears to teach cycling allocation of registers in a pool of registers as some of successively identified registers in the emulated sequence of instructions ***(page 4, code example below paragraph 2)***.

19.7. Regarding claim 29, Wall appears to teach producing a history of register allocation information ***(page 1, Abstract, paragraph 3; and page 12, section 2.4.4. Profiling)***.

19.8. Regarding claim 29, Wall appears to teach changing register parameters of the emulated sequence of instructions ***(page 1, Abstract, paragraph 3; and page 12, section 2.4.4. Profiling)***.

19.9. The art of Wall and the art of Altman are analogous art because they both include the problem of profiling a program execution and using the profile information to optimize the program.

19.10. The motivation to use the art of Wall with the art of Altman would have been obvious because of the benefit recited in Wall that benchmark program speedup was 10 to 25 percent ***(page 1, Abstract, paragraph 3)***.

19.11. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Wall with the art of Altman to produce the claimed invention.

20. Claim 30 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman and common knowledge in the art, as applied to claim 26 above, in view of Wall (Wall, David W.; "Global Register Allocation at Link Time", October 28, 1986, www.hpl.hp.com/techreports/Compaq-DEEC/WRL-86-3.html).

20.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

20.2. The art of Wall is directed to optimizing register allocation, including using profile information to perform optimization ***(page 1, Abstract)***.

20.3. Altman appears to teach an emulation code generator generating the emulated sequence of instructions executable a first instruction set from the original sequence of instructions executable with a different second instruction set ***(page 40, title; and abstract directly beneath the title; and left-side column, paragraph 3, definition of "binary translation")***.

20.4. Altman appears to teach modifying the emulated sequence of instructions in response to at least the historical register usage information ***(page 41, section "Dynamic optimization")***.

20.5. Altman does not specifically teach generating historical register usage information regarding register status during execution of the emulation sequence of instructions.

20.6. Wall appears to teach generating historical register usage information regarding register status during execution of the emulation sequence of instructions ***(page 1, Abstract, paragraph 3; and page 12, section 2.4.4. Profiling)***.

20.7. The art of Wall and the art of Altman are analogous art because they both include the problem of profiling a program execution and using the profile information to optimize the program.

20.8. The motivation to use the art of Wall with the art of Altman would have been obvious because of the benefit recited in Wall that benchmark program speedup was 10 to 25 percent ***(page 1, Abstract, paragraph 3)***.

20.9. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Wall with the art of Altman to produce the claimed invention.

21. Claim 31 is rejected under 35 U.S.C. 103(a) as being unpatentable over Altman and common knowledge in the art, in view of Conte (Conte, Thomas M.; Patel, Burzin A.; Cox, J. Stan; "Using Branch Handling Hardware to Support Profile-Driven Optimization", 1994, Proceedings of the 1994 27th annual international symposium on microarchitecture).

21.1. The art of Altman is directed toward translating machine instructions for one computer into machine instructions to run on a second computer, including profiling and dynamic optimization ***(pages 40 - 41)***.

21.2. The art of Conte is directed to using branch handling hardware to support profile-driven optimization ***(page 1, Title)***.

21.3. Altman appears to teach an emulation code generator generating the emulated sequence of instructions executable a first instruction set from the original sequence of instructions executable with a different second instruction set ***(page 40, title; and abstract directly beneath the title; and left-side column, paragraph 3, definition of "binary translation")***.

21.4. Altman does not specifically teach generating historical branch prediction dynamic execution information regarding likelihood of branches taken during execution of the emulation sequence of instructions.

21.5. Altman does not specifically teach generating a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group and is dependent upon a combined effect of the branch predictions of the members of the group.

21.6. Conte appears to teach generating historical branch prediction dynamic execution information regarding likelihood of branches taken during execution of the emulation sequence of instructions ***(pages 3 – 4, section 3 Using Branch Prediction Hardware for Profiling; and page 2, section 2 Branch Prediction and Profiling)***.

21.7. Conte appears to teach generating a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group and is dependent upon a combined effect of the branch predictions of the members of the group ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***.

21.7.1. Regarding ***(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph; and figure 2)***; it would have been obvious to generate a branch prediction likelihood code for a group of branches that may be different from any branch prediction of the members of the group and is dependent upon a combined effect of the branch predictions of the members of the group.

21.8. The art of Conte and the art of Altman are analogous art because they both contain the problem of profiling and optimization.

21.9. The motivation to use the art of Conte with the art of Altman would have been obvious given the benefit recited in Conte that using the specified branch prediction method produces a dramatic effect in achieving 96% accuracy in branch prediction

(page 2, section 2.1 Contemporary branch handling mechanisms, second paragraph);

21.10. Therefore, as discussed above, it would have been obvious to the ordinary artisan at the time of invention to use the art of Conte with the art of Altman to produce the claimed invention.

Conclusion

22. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Russell L. Guill whose telephone number is 571-272-7955. The examiner can normally be reached on Monday – Friday 9:00 AM – 5:30 PM.

23. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kevin Teska can be reached on 571-272-3716. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Any inquiry of a general nature or relating to the status of this application should be directed to the TC2100 Group Receptionist: 571-272-2100.

24. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

RG

KEVIN J. TESKA
SUPERVISORY
PATENT EXAMINER